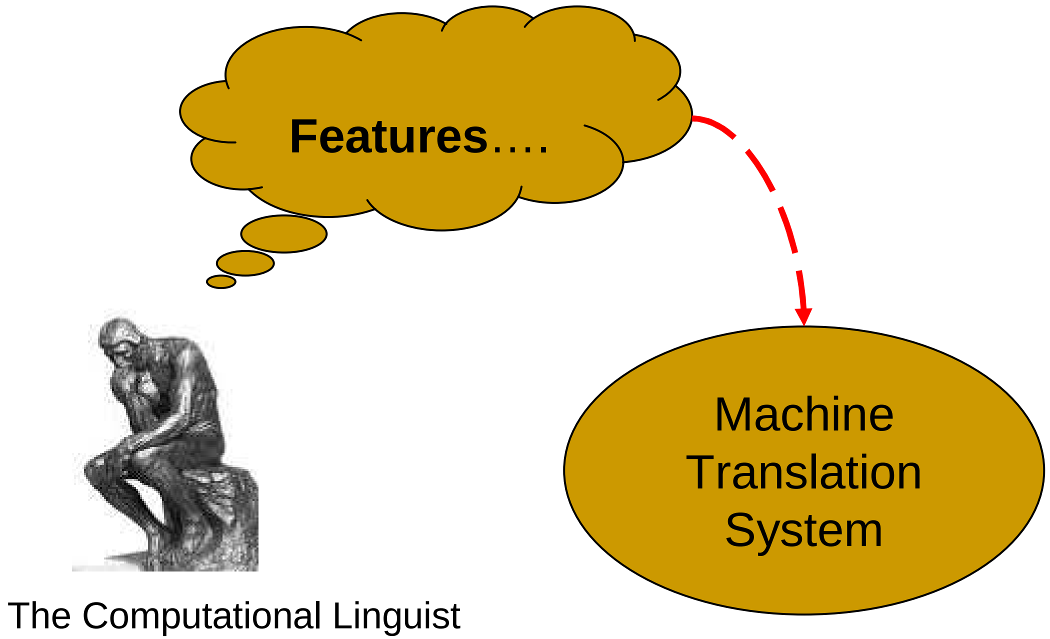
N-best Reranking by Multitask Learning

Kevin Duh, Katsuhito Sudoh, Hajime Tsukada, Hideki Isozaki, Masaaki Nagata

NTT Communication Science Laboratories

Our Goal

Incorporate ***millions of features*** into MT without **overfitting!**



Main Ideas

1. Some features are just very sparse
2. Overfitting is inevitable for conventional training
3. But multitask learning can help by discovering lower-dimensional feature space

Outline

1. **WHY**: Motivations
   - The challenge of sparse features
2. **HOW**: Proposed training algorithm
3. **WHAT**: Reranking experiments
4. Conclusions

Background

- Goal: given *f*, score translations *e* based on:

$$\hat{e} = \arg\max_{e \in N(f)} \mathbf{w}^T \cdot \mathbf{h}(e, f)$$

N-best List

Trained weights

Features

- We're interested in systems employing *millions of features*

Note: Here we focus on N-best reranking but extension to 1st-pass training is possible

Sparse features for MT

- [Watanabe2007] proposed heavily-lexicalized features, e.g.

$$h(e,f) = \begin{cases} 1 & \text{if foreign word ``Monsieur'' and English word ``Mr.'' co-occur in } e,f \\ 0 & \text{otherwise} \end{cases}$$

Never used if input sentence does not contain "Monsieur"

$$h(e,f) = \begin{cases} 1 & \text{if English trigram ``Mr. Smith said'' occurs in } e \\ 0 & \text{otherwise} \end{cases}$$

Many reordering possibilities
→ many potential features
"said Smith Mr.", "Smith Mr. said",..

Why does overfitting occur?

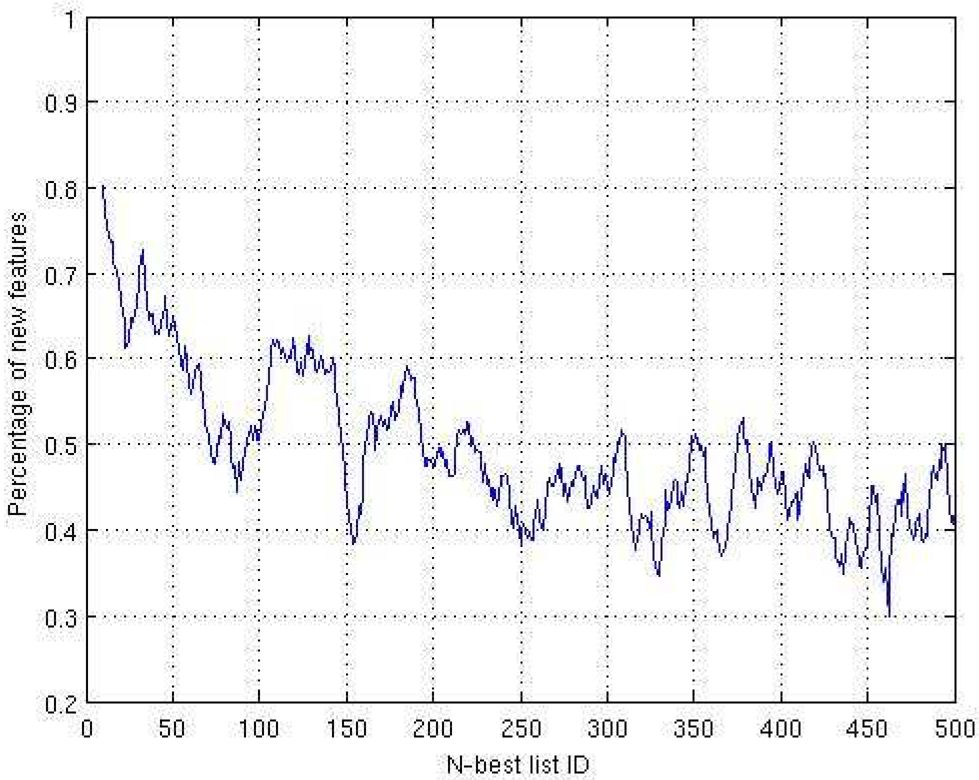
Because there exist
very little feature overlap
between any two N-best lists.

Visualizing feature overlap (or lack thereof)

Feature Growth Rate

Definition: ratio of new-feature to active feature

In the limit, 45% of active features are never seen before!



Conditions for this long-tail behavior

-Feature templates are heavily-lexicalized

-Input (f) has high variability

-Output (e) has high variability

Outline

1. WHY: Motivations
2. HOW: Proposed training algorithm
   - What is multitask learning
   - How N-best can be viewed as multitask problem
3. WHAT: Reranking experiments
4. Conclusions

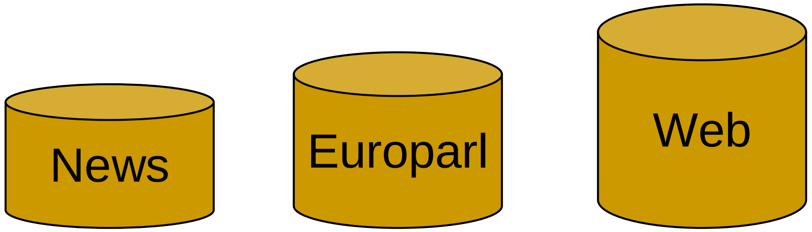
What is Multitask Learning?

A set of machine learning techniques
for exploiting **heterogeneous** training data

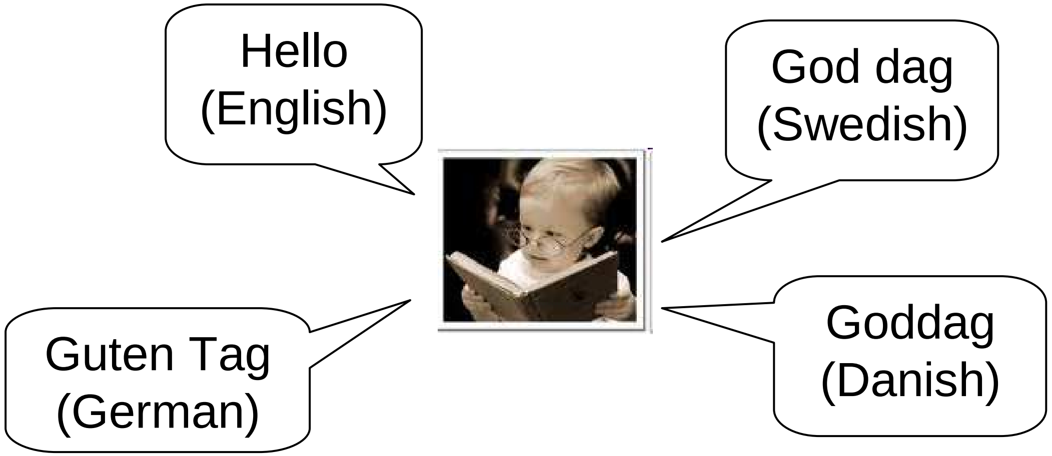
- Contrasts with i.i.d. assumption of traditional setup
- Instead assumes some underlying commonality

Examples of "Tasks"

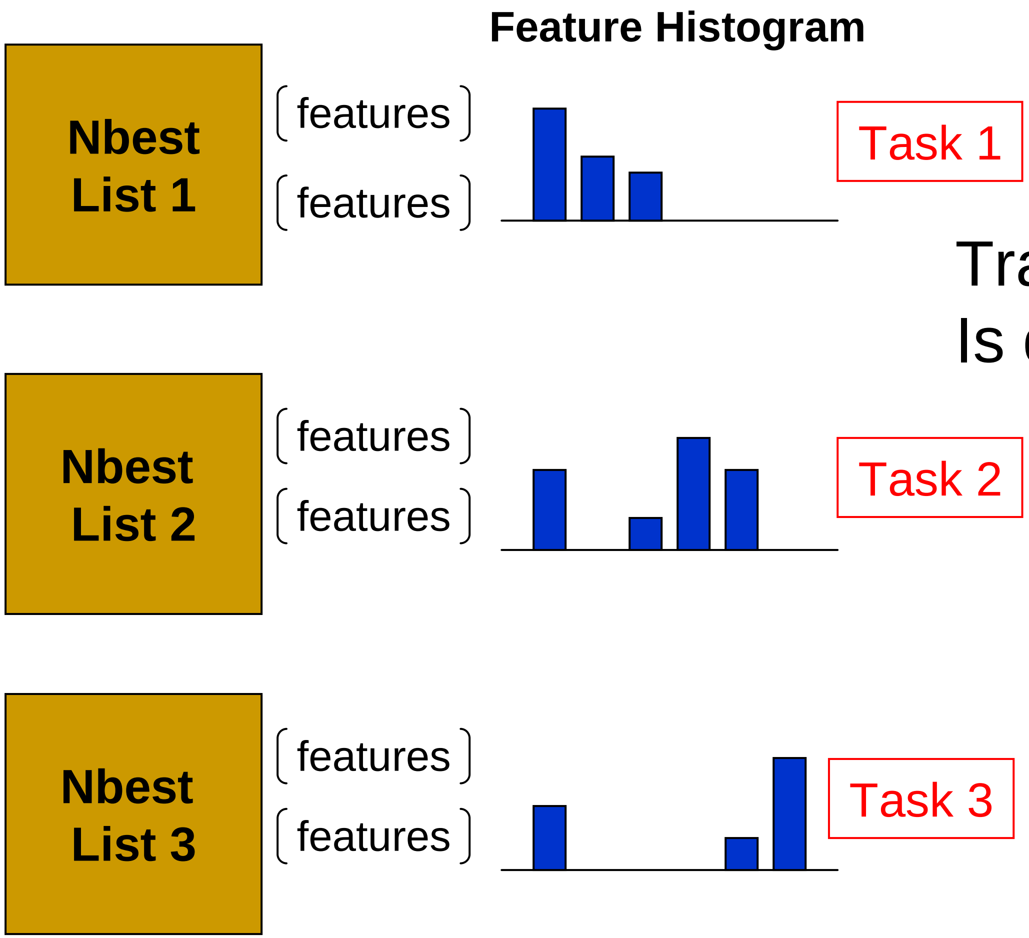
- Multiple domains:



- Multiple related problems:



N-bests with sparse features can be viewed as a Multitask problem



Train a single weight **w** ?
Is data i.i.d. across N-bests?

NO!
Data is heterogenous.
Treat as multitask!

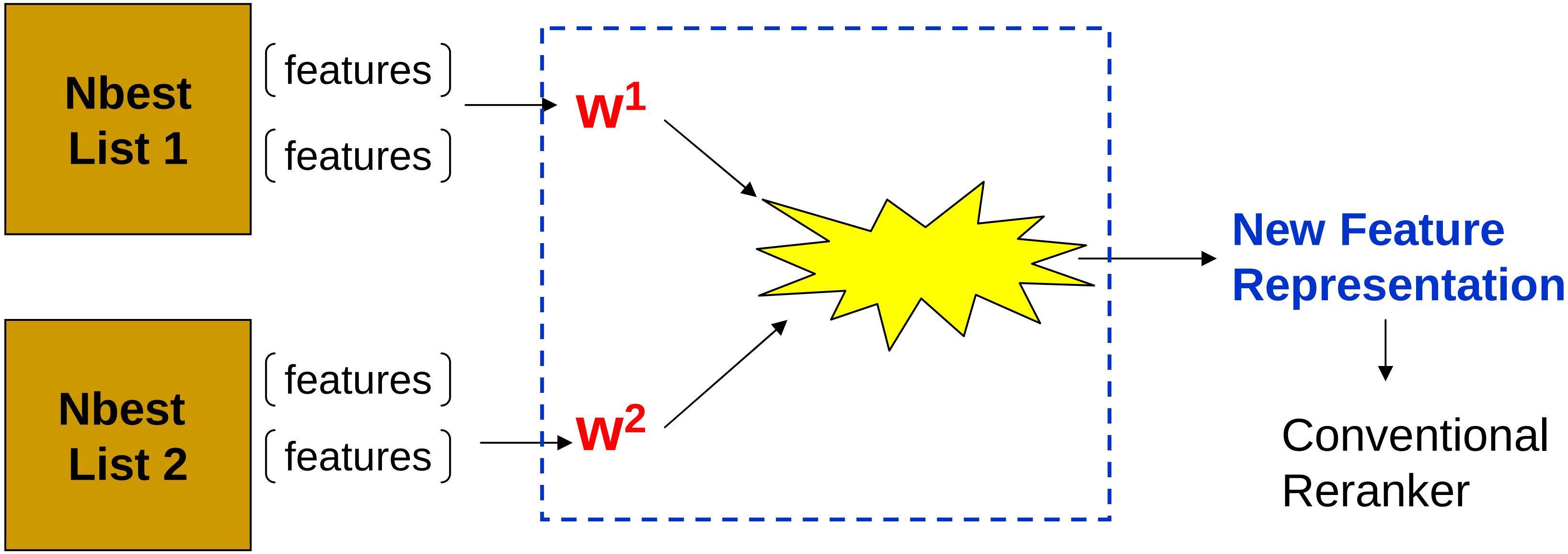
Our Meta-Algorithm

STEP 1: Train weights independently for each N-best

STEP 2: Find commonality among weights (and iterate)

←Plug in your favorite Multitask Learning method

STEP 3: Train conventional reranker on discovered common features



L1/L2 Joint Regularization

(one example multitask learning method)

$$\arg\min_{w^1,w^2,..,w^I} \sum_{i=1}^{I} Loss(w^i, nbest^i) + \lambda \|W\|_{1,2}$$

$\|W\|_{1,2}$ computed by
1. Stacking the weights into a matrix
2. Take L2 norm on columns, then L1 norm on result

Effect: encourage sharing of features

Exercise: which is the better solution?

$$\mathbf{W_a}: \begin{bmatrix} 4 & 0 & 0 & 3 \\ 0 & 4 & 3 & 0 \end{bmatrix} \qquad \mathbf{W_b}: \begin{bmatrix} 4 & 3 & 0 & 0 \\ 0 & 4 & 3 & 0 \end{bmatrix}$$

$$4 \quad 4 \quad 3 \quad 3 \rightarrow 14 \qquad\qquad 4 \quad 5 \quad 3 \quad 0 \rightarrow 12$$

Many multitask methods are available!

Joint Regularization:

- L1/L2 [Obozinski09, Argyriou08]
- L1/L-infinity [Quattoni09]

Bayesian Prior: [Daume09, Finkel09]

$$\sum_i ||\mathbf{w}^i - \mathbf{w}^{avg}||_2$$

Shared Feature Subspace:

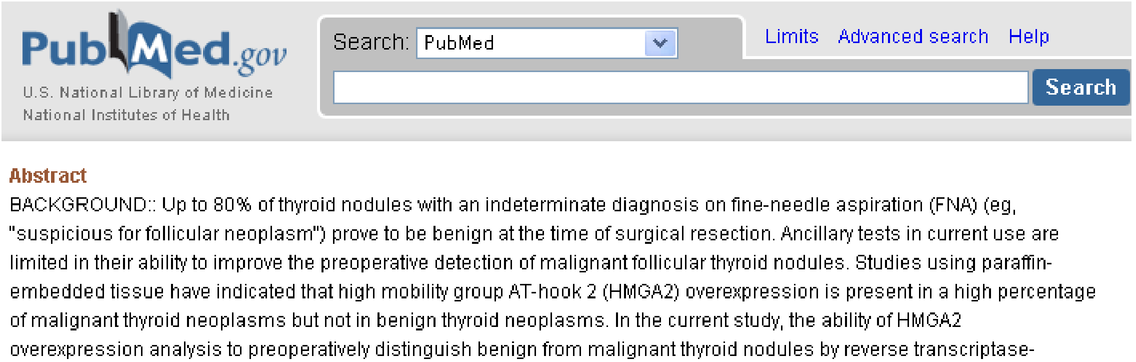
- SVD-based [Ando05]
- Neural network [Caruana97]
- Deep Learning [Collobert08]

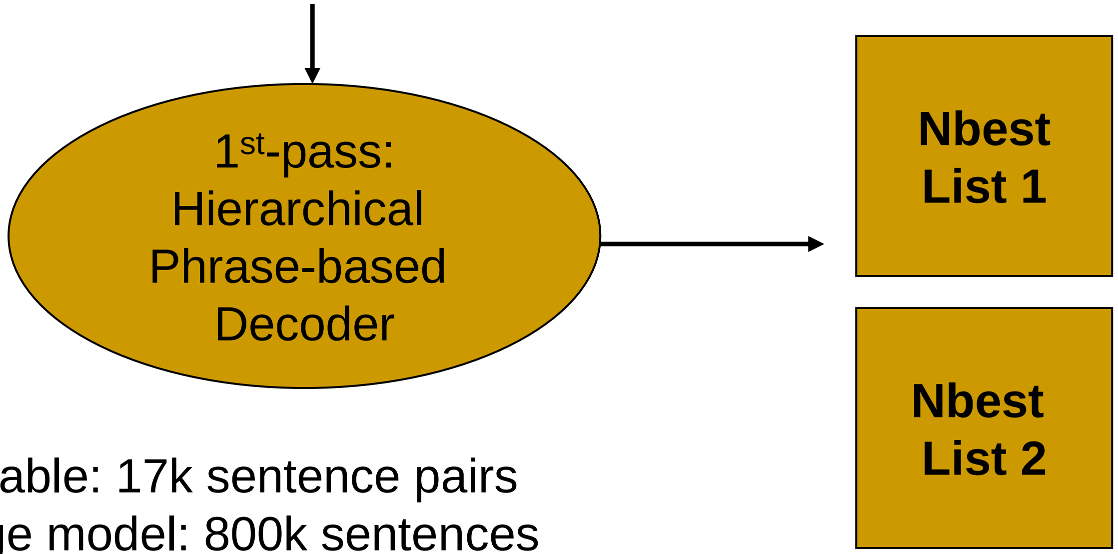
Outline

1. WHY: Motivations
2. HOW: Proposed training algorithm
3. WHAT: Reranking experiments
   - Data
   - Results
4. Conclusions

Data

English→Japanese translation of PubMed abstracts





N=100
500 lists for train
100 lists for tune
500 lists for test

Phrase table: 17k sentence pairs
Language model: 800k sentences

Experiment comparison

- What is best feature representation?

Baselines:

1. Original Feature Representation
2. Feature selection by L1 regularization

vs.

Features discovered by Multitask:

1. Joint Regularization (L1/L2)
2. Shared Subspace (SVD)

- Specifics:
  - Base reranker is RankSVM, similar to [Shen04]
  - Original: 2.4 million features
  - Tune multitask feature dimension: {250,500,1000}

Results

Feature Representation	No. of features	Train BLEU	Test BLEU
First pass system features	20	29.5	28.5
Baseline 1: Original Sparse Features	2.4M	36.9	**28.6**
Baseline 2: Original, with L1 regularization	1200	36.5	28.5
Oracle	--	36.9	36.9
Multitask 1: Joint Regularization (L1/L2)	250	31.8	**28.9**
Multitask 2: Shared Subspace (SVD)	1000	32.9	**29.1**
Feature Threshold (occurs in 10+ lists)	60k	35.8	**29.0**
+ Multitask 1: Joint Regularization	60.25k	36.1	**29.4**
+ Multitask 2: Shared Subspace	61k	36.2	**29.6**

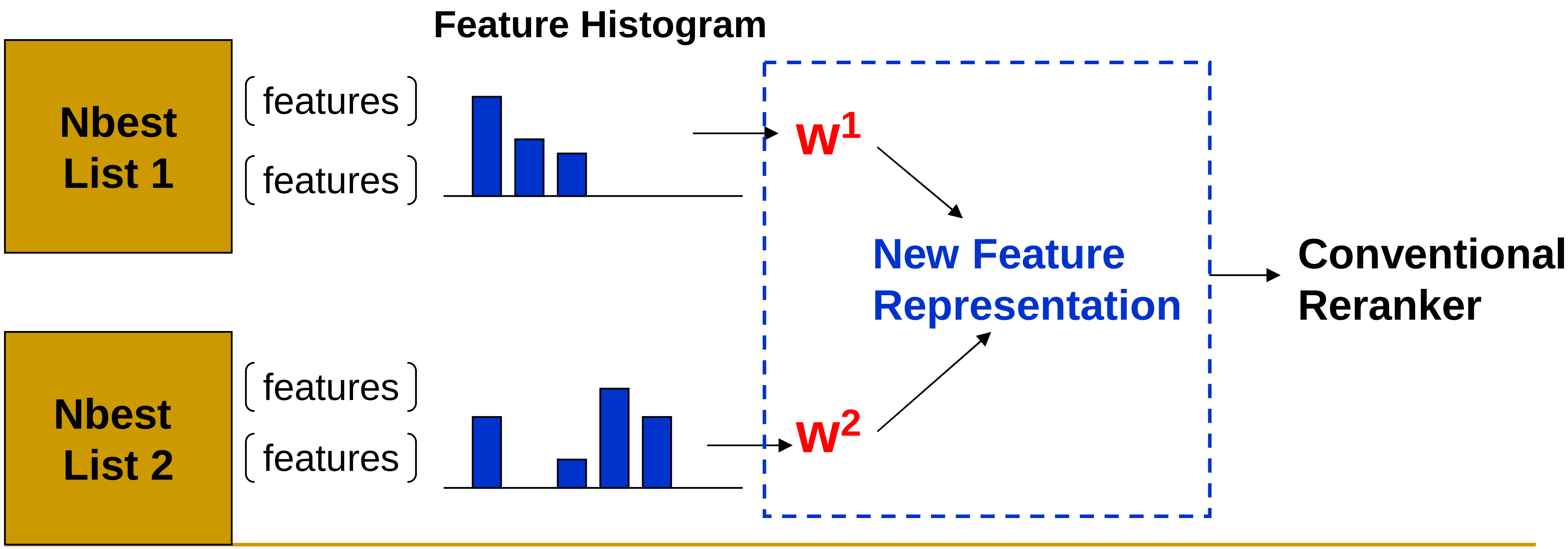
Improvements in **red** are statistically significant by bootstrap test ($p<0.05$)

Outline

1. WHY: Motivations
2. HOW: Proposed training algorithm
3. WHAT: Reranking experiments
4. Conclusions (2 slides)

Contributions

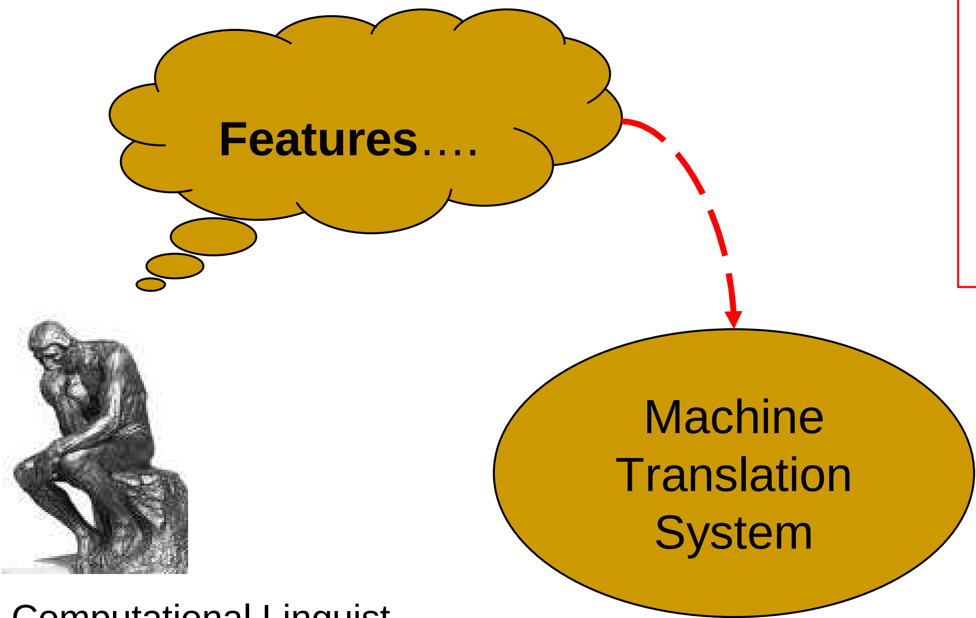
- N-best Lists with sparse features may be cast as multitask problem
- Proposed meta-algorithm uses multitask methods to learn better features for reranking



Final Words

MORE FEATURES IS THE WAY TO GO:

Translation is a delicate process requiring many fine-grained knowledge



The Computational Linguist

But we must avoid overfitting:

1. Careful definition of features: e.g. [Chiang09,Marton08]
2. Feature mining [This work]

Thanks! Questions? Suggestions?

- **Citations**:

  - [Ando05]: A framework for learning predictive structures from multiple tasks, JMLR
  - [Argyriou08]: Convex multitask feature learning, MLJ
  - [Chiang09]: 11,001 new features for SMT, NAACL
  - [Collobert08]: A unified architecture for NLP: deep neural networks with multitask learning, ICML
  - [Daume09]: Bayesian multitask learing with latent hierarchices, UAI
  - [Marton08]: Soft syntactic constraints for hierarchical phrase based translation, ACL
  - [Finkel09]: Hierarchical Bayesian domain adaptation, NAACL
  - [Quattoni09]: An efficient projection for L1-Linf regularization, ICML
  - [Shen04]: Discriminative reranking for MT, NAACL
  - [Watanabe07]: Online large margin training for SMT, EMNLP

- **Acknowledgments**:

  - We thank Jun Suzuki, Shinji Watanabe, Albert Au Yeung and the three reviewers for their valuable comments!

Feature Representation	#Feature	Train BLEU	Test BLEU	Test PER
(baselines)				
First pass	20	29.5	28.5	38.3
All sparse features (Main baseline)	2.4M	36.9	28.6	38.2
All sparse features w/ ℓ_1 regularization	1200	36.5	28.5	38.6
Random hash representation	4000	33.0	28.5	38.2
(multitask learning)				
Unsupervised FeatureSelect	500	32.0	**28.8**	**37.7**
Joint Regularization	250	31.8	**28.9**	**37.5**
Shared Subspace	1000	32.9	**29.1**	**37.3**
(combination w/ high-frequency features)				
(a) Feature threshold $x > 100$	3k	31.7	27.9	38.2
(b) Feature threshold $x > 10$	60k	35.8	29.0	37.9
Unsupervised FeatureSelect + (b)	60.5k	36.2	**29.3**	**37.6**
Joint Regularization + (b)	60.25k	36.1	**29.4**	**37.5**
Shared Subspace + (b)	61k	36.2	**29.6**	**37.3**
Oracle (best possible)	–	36.9	36.9	33.1

Open Questions

- Interactive feature engineering?
- Different partition of tasks?
- Multitask on lattices or larger N-bests?
- Comparison to online learning?

A Bayesian perspective

1^{st} Pass Decoder P(e|f) generates data conditioned on f

- f is task-specific "parameter"
- P(e|f) is common across tasks